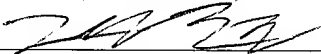


PATENT
5181-83600
P5882

"EXPRESS MAIL" MAILING LABEL
NUMBER EL822012121US
DATE OF DEPOSIT APRIL 25, 2001
I HEREBY CERTIFY THAT THIS PAPER OR
FEE IS BEING DEPOSITED WITH THE
UNITED STATES POSTAL SERVICE
"EXPRESS MAIL POST OFFICE TO
ADDRESSEE" SERVICE UNDER 37 C.F.R. §
110 ON THE DATE INDICATED ABOVE
AND IS ADDRESSED TO COMMISSIONER
FOR PATENTS, WASHINGTON, D.C. 20231


Derrick Brown

MAINTAINING FABRIC DEVICE CONFIGURATION
THROUGH DYNAMIC RECONFIGURATION

by

Hyon T. Kim

Robert C. Kowert
Conley, Rose & Tayon, P.C.
P.O. Box 398
Austin, Texas 78767-0398
Phone (512) 476-1400

BACKGROUND OF THE INVENTION

1. Field of the Invention

- 5 **[0001]** This invention relates to maintaining a host system's configuration of devices attached to a fabric in a storage network.

2. Description of the Related Art

- 10 **[0002]** Storage area networks, also referred to as SANs, are dedicated networks that connect one or more systems to storage devices and subsystems. Today, fibre channel is one of the leading technologies for SANs. In general, fibre channel encompasses three networking topologies: point-to-point, loop, and fabric. In a point-to-point topology, a fibre channel host adapter in a system is typically connected to a single fibre channel storage subsystem. In a fibre channel loop network, also called an arbitrated loop, the loop is constructed by connecting nodes together in a single logical ring. Loops can be constructed by connecting nodes through a fibre channel hub in a star-wired topology or by connecting them together in a connected physical loop from node to node. In a fibre channel fabric topology, the storage networks are constructed with network switches. A fabric can be composed of a single switch or multiple switches. Ports on fabric networks connect nodes to switches on low-latency, point-to-point connections.

- 25 **[0003]** The nodes connected in the loop and fabric topologies refer to "network nodes" and can be any entity that is able to send or receive transmissions in a fibre channel network. For example, a node can be a computer system, a storage device/subsystem, a storage router/bridge that connects SCSI equipment, a printer, a scanner, or any other equipment, such as data capture equipment. The ANSI X3.272-1996 specification entitled "FC-AL, Fibre Channel Arbitrated Loop" and the ANSI X3.T11 Project 1133-D specification entitled "FC-AL-2, Fibre Channel Arbitrated Loop" describe examples of fibre channel loop topologies in further detail. The ANSI X3.T11

Project 959-D specification entitled “FC-SW Fibre Channel Switch Fabric” describes an example of a fibre channel fabric in further detail. Note that the most recent versions of these and related specifications may be obtained from the T11 technical committee of the National Committee for Information Technology Standards (NCITS).

5

[0004] For point-to-point topologies and loop topologies, device drivers executing on a host computer perform device discovery at host boot-up to determine locally connected devices. The discovered devices are configured to be accessible to applications running on the host by creating a node for each device within the host. These types of nodes are referred to as operating system device nodes. Each node functions as an internal (to the host) representation of an attached device and provides a communication path to the device. Since the number of devices attached to a host in point-to-point and loop topologies are limited, the discovery process may be completed within an acceptable time frame. Due to the number of devices capable of being attached to a fabric, discovering all fabric devices available to the host at host boot-up may not be feasible. Additionally, a host computer may desire to maintain its fabric device configuration across reboots and adjust for changes in the fabric.

SUMMARY OF THE INVENTION

[0005] Methods, systems, and programs for maintaining a fabric-attached device configuration in regard to fabric state changes for a host system connected to a fabric are described for various embodiments of the present invention. An event may be received indicating a fabric state change for one or more host adapter ports. In response to the event, the host system's fabric device configuration may be dynamically changed. Dynamically changing the host system's fabric device configuration may include bringing online or taking offline one or more fabric devices for the one or more host adapter ports for the host system.

[0006] In some embodiments, a configuration file may be stored specifying an action to be taken for various types of events. Upon receiving indicating a fabric state change, the configuration file may be accessed to determine the fabric device configuration action to be taken for the type of fabric state change indicated by the event.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] Figure 1 illustrates a host computer attached to a fabric and one or more local devices;

5

[0008] Figure 2 illustrates an example of a storage area network (SAN) suitable for implementing various embodiments of the present invention;

[0009] Figure 3 is a flowchart illustrating an on-demand node creation process according to an embodiment of the present invention;

10

[0010] Figure 4 illustrates an example of a storage network suitable for implementing various embodiments of the present invention;

[0011] Figure 5 is an illustration of a host computer coupled to a fabric according to one embodiment of the present invention;

15

[0012] Figure 6 is a flowchart illustrating a device discovery process, according to one embodiment of the present invention;

20

[0013] Figure 7 is a flowchart illustrating an on-demand device node creation process, according to one embodiment of the present invention;

[0014] Figure 8 is an illustration of a system coupled to a fabric according to an embodiment of the present invention;

25

[0015] Figure 9 is a flowchart of an on-demand node creation process for fibre channel fabrics according to an embodiment of the present invention;

[0016] Figure 10 is a flow chart illustrating a method for onlining selected fabric devices on-demand from a list of fabric devices, according to one embodiment of the present invention;

5 **[0017]** Figure 11 is a flowchart illustrating a mechanism for dynamically updating a persistent repository to reflect a changes a in the fabric, according to one embodiment of the present invention;

[0018] Figure 12 is a flowchart illustrating a mechanism to allow a host's fabric
10 device configuration to persist across reboots and shutdowns, according to one embodiment of the present invention;

[0019] Figure 13 is a diagram illustrating components for providing dynamic node reconfiguration in response to fabric state change events, according to one embodiment of
15 the present invention;

[0020] Figure 13A is an illustration of modules which may be included in a fabric driver according to one embodiment of the present invention;

20 **[0021]** Figure 14 is a flow chart illustrating an event driven node configuration process in response to dynamic changes in fabric connectivity, according to one embodiment of the present invention;

[0022] Figure 15 illustrates a system including a host adapter configuration file,
25 according to one embodiment of the present invention;

[0023] Figure 16 is a flowchart illustrating an event driven configuration process for an event in which a host adapter loses connectivity to the fabric, according to one embodiment of the present invention;

30

[0024] Figure 17 is a flowchart illustrating an event driven configuration process for an event in which a host adapter acquires or regains connectivity to the fabric, according to one embodiment of the present invention; and

5 **[0025]** Figure 18 illustrates a flowchart for an event driven configuration in regard to a fabric state change of the connection of a new fabric device, according to one embodiment of the present invention.

[0026] While the invention is described herein by way of example for several
10 embodiments and illustrative drawings, those skilled in the art will recognize that the invention is not limited to the embodiments or drawings described. It should be understood, that the drawings and detailed description thereto are not intended to limit the invention to the particular form disclosed, but on the contrary, the intention is to cover all modifications, equivalents and alternatives falling within the spirit and scope of the
15 present invention as defined by the appended claims.

DETAILED DESCRIPTION OF EMBODIMENTS OF THE INVENTION

[0027] Figure 1 illustrates a host system 108 attached to a fabric 104, which is suitable for implementing various embodiments of the present invention. The host system 108 may include at least one central processing unit (CPU) or processor 102. The CPU 102 may be coupled to a memory 112. The memory 112 is representative of various types of possible memory media (also referred to as “computer readable media”): for example, hard disk storage, floppy disk storage, removable disk storage, flash memory or random access memory (RAM). The terms “memory” and “memory medium” may include an installation medium, e.g., a CD-ROM or floppy disk, a computer system memory such as DRAM, SRAM, EDO RAM, SDRAM, DDR SDRAM, Rambus RAM, etc., or a non-volatile memory such as a magnetic media, e.g., a hard drive, or optical storage. The memory medium may include other types of memory as well, or combinations thereof. In addition, the memory medium may be located in a first computer in which the programs are executed, or may be located in a second different computer which connects to the first computer over a network. In the latter instance, the second computer provides the program instructions to the first computer for execution.

[0028] As shown in Figure 1 the memory 112 permits two-way access: it is readable and writable. The memory 112 may store instructions and/or data which implement all or part of the system and method described in detail herein, and the memory 112 may be utilized to install the instructions and/or data. In various embodiments, the host system 108 may take various forms, including a personal computer system, desktop computer, laptop computer, palmtop computer, mainframe computer system, workstation, network appliance, network computer, Internet appliance, personal digital assistant (PDA), embedded device, smart phone, television system, or other suitable device. In general, the term “computer system” may be broadly defined to encompass any device having a processor which executes instructions from a memory medium.

[0029] The host system 108 may be coupled to a fabric 104, which may provide access to a plurality of fabric attached devices, such as persistent storage devices or other

computer peripheral devices. The CPU 102 may acquire instructions and/or data through an input/output interface 110. Through the input/output interface 110, the CPU 102 may also be coupled to one or more local devices 103, such as local input/output devices (video monitors or other displays, track balls, mice, keyboards, etc.) local storage devices (hard drives, optical storage devices, etc.), local printers, plotters, scanners, and any other type of local I/O devices for use with a host system 108. Some local devices may be referred to as direct attach devices. Input/output interface 110 may include host adapters 111a and 111b for coupling to local devices 103 and fabric 104 respectively. Host adapters 111a and 111b may be fibre channel adapters (FCAs). In one embodiment, one or more of the local devices may be included in the host system 108, such as in expansion slots of the host system 108. In one embodiment, one or more of the local devices may be externally connected to the host system 108.

[0030] The host system 108 may be operable to execute one or more computer programs. The computer programs may comprise operating system or other system software, application software, utility software, Java™ applets, and/or any other sequence of instructions. Typically, an operating system (OS) performs basic tasks such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk, and controlling peripheral devices such as disk drives and printers. Application software runs on top of the operating system and provides additional functionality. In one embodiment, the OS may be based on Sun Microsystems's Solaris™ operating system. The computer programs may be stored in a memory medium or storage medium such as the memory 112, or they may be provided to the CPU 102 through the fabric 104 or I/O interface 110.

[0031] Figure 2 illustrates an example of a SAN coupled to host computers 108A and 108B. The SAN shown in Figure 2 includes a fabric interconnect 105 coupled to hard drives 210A, 210B, and 210C, tape drive 211, and optical drive 212. Hard drives 210A, 210B, and 210C, tape drive 211, and optical drive 212 are also referred to as fabric devices. It should be noted that the number and types of hosts and devices shown in Figure 2 are for illustration purposes only, and the actual number and types of hosts

and/or devices in a SAN may vary. Fabric interconnect 105 may include one or more switches which are connected to the network nodes with an optical, copper cable, or other type of cable or interconnect medium.

5 **[0032]** Figure 4 illustrates an example of a storage network which includes a direct attached private loop 306 and a fabric 410. It should be noted that storage networks may be configured in a variety of different ways and many include one or more direct attach devices, SANs, and/or network attach (NAS) devices. Furthermore, it should be noted that fabrics and/or SANs are not limited to fibre channel technologies and architectures
10 but may include various types of technologies. For example, some or all of a SAN may be based on the InfiniBand™ Architecture or iSCSI (SCSI over IP).

15 **[0033]** Host adapter 304 couples host system 402a to private loop 306 and adapters 404a and 404b couples host system 402b to fabric 410. Note that host adapters 304, 404a and 404b may be separate host bus adapter cards, for example. In other embodiments, host adapters 304, 404a and 404b may each refer to a separate host adapter port. Coupled to private loop 306 are one or more direct attach devices 308. Direct attach device(s) 308 are considered local to host system 402a.

20 **[0034]** The host system 402b is coupled to fabric 410 via host adapter 404c. Fabric 410 includes fibre channel switches 412 which are coupled to multiple fabric devices 408. Each fibre channel switch may connect to various fibre channel topologies such as point-to-point fibre channel connections or fibre channel loops. Each switch may also connect to one or more other fibre channel switches. The fabric devices 408 may be various
25 storage devices such as hard disk drives, optical drives, tape drives, etc. In some embodiments fabric devices 408 may be any type of I/O device such as storage devices, printers, scanners, etc. as are used in conjunction with computer systems.

30 **[0035]** Due to the large number of devices which may be present in fabric 410, the time required for a host system 402 to discover and online all of the devices available

through fabric 410 may be impractical. Furthermore, it may be unlikely that host system 402 actually needs to communicate with all of the various fabric devices 408. The term “online” may be used herein to refer to creating a node to provide a communication mechanism or path to a device.

5

[0036] Figure 5 illustrates a host system 508 coupled to a fabric 510 according to one embodiment of the present invention. The fabric 510 may be implemented with one or more switches coupled to one or more storage devices or subsystems. Furthermore, it should be noted that fabric 510 is not limited fibre channel fabrics but may be extended to any type of switched storage network. A fabric driver 504 provides an interface between host system 508 and fabric 510. A persistent repository 506 is a data structure that stores information on the current status of the fabric devices. An administration application 502 is a software program running on host system 508 and accessible to a system administrator. Administration application 502 provides a mechanism for discovering fabric devices on-demand with user input thereby eliminating the need to discover all fabric devices accessible to a host system at one time. As such, nodes can be created only for selected fabric devices (i.e., a subset of available fabric devices) during a discovery process. It should be noted that if a node is already created for a selected fabric device, it may not be necessary to re-create the node.

20

[0037] During a discovery process, administration application 502 may query a fabric driver 504 for a list of devices visible to the host system 508. Fabric driver 504 provides an interface for the host system to the fabric. Fabric driver 504 may be part of the operating system for the host system and may include one or more modules for handling various functions required to interface the host system to the fabric such as protocol handling and transport layer operations. In one embodiment of the present invention, the fabric driver 504 may be a Solaris kernel module or modules. The fabric driver 504 may provide the requested list of devices to the administration application 502. A subset of these devices may then be selected through administration application 502 and brought online by fabric driver 504 so that the subset of devices are accessible from host system

30

508. Onlining the subset of devices may include the creation of a node within the operating system for each device wherein the node provides a reference for applications or other processes in host system 508 to reference a corresponding device in fabric 510. Thus, a node provides a path for an application or process running on the host system to communicate with one of the fabric devices. Thus, when a fabric device is online, a communication path may be established between the host system 508 and the discovered fabric device.

[0038] Administration application 502 thus provides a mechanism to select and online only a subset of the visible fabric devices. Also administration application 502 may be run outside of the boot process so that device discovery and online operations do not increase the host system boot time. The administration application 502 may be run on-demand so that fabric devices may be selected and brought online on demand. For one embodiment of the present invention, a system administrator may use administration application 502 to request a list of fabric devices available through one or more host adapter I/O ports of host system 508. Administration application 502 may include a command line interface or a graphical user interface (or both) for displaying the list to the system administrator. Through this interface the system administrator may then select a desired subset of the listed devices and request that the selected devices be brought online.

[0039] The fabric driver 504 provides APIs for the administration application 502 to make queries in order to obtain a list of devices connected to one or more host adapter ports. In one embodiment a transport layer of the fabric driver 504 may execute the query irrespective of the interconnect topology of the host adapter ports. For example, the fabric driver may obtain the list of devices connected to a fibre channel switched fabric by querying a fabric name server. The fabric name server may be located within a fabric switch or distributed across the fabric switches and maintains information about the various fabric devices. The fabric name server may include a database of objects. Each fabric attached device may register or query useful information in the name server,

including, name, address, class of service capability of other participants, etc. The fabric driver may also provide an API for obtaining a list of direct attach devices for a host. For example, the fabric driver may obtain the loop map for a host system's private loop topology.

5

[0040] For direct attach devices, e.g. private loop topologies, operating system device nodes may be created during driver attach (e.g., when the fabric driver is loaded during a reboot) for all devices visible through a direct attach port. In some embodiments, operating system device nodes are only created for direct attach devices that support a particular protocol (e.g., FCP/SCSI).

10

[0041] For fabric topologies connected to the host system 508, operating system device nodes may not be created until an on-demand request is made by the administration application 502. As discussed above, upon such a request from the administration application 502, the fabric driver 504 provides a list of devices visible through one or more fabric host adapter ports. The fabric driver 504 may obtain the list by accessing a fabric name server. The list of fabric devices may be provided by the administration application 502 to a system administrator, for example, through a graphical user interface. The system administrator may use the administration application 502 to select and online particular devices which are desired to be used by the host system 508. The administrator may also use administration application 502 to offline any devices which are no longer needed. Offlining may be removing access to a storage device form the host operating system by removing the node. The list of devices displayed to the administrator by the administration application 502 may include the information returned about the fabric devices by the fabric name server. The administration application 502 may request the fabric driver 504 to online or offline fabric devices as indicated by a system administrator using the application's user interface. In some embodiments, the administration application 502 may make requests to fabric driver 504 to obtain fabric device information and/or to online/offline fabric devices without the involvement of a system administrator. For example, certain events

15

20

25

30

or requests from other processes may trigger administration application 504 to online a fabric device(s).

5 [0042] The persistent repository 506 may be stored in the host system 508, or in some central local accessible to host system 508, indicating the current fabric devices which have been onlined (e.g., devices for which an operating system device node has been created). The information stored in the persistent repository 506 may be used so that the configuration of devices online for the host system 508 persists across reboots and shutdowns. For example, when the host system 508 is rebooted the persistent repository
10 may be read to determine which devices were online before the reboot and the fabric driver may be requested to online these same devices again.

15 [0043] The persistent repository 506 may be dynamically updated to reflect the state of the fabric devices. For example, if a fabric device which is currently online for the host system 508 is disabled on the fabric (for example, a hard drive fails or is removed from the fabric), fabric driver 504 may generate an event causing the persistent repository to be updated to reflect that the device is now offline. Similarly, if the same device later is restored on the fabric, the device may be onlined again (e.g. in response to an event) for the host system 508 and the persistent repository dynamically updated to reflect the
20 onlined status.

[0044] Although fabric 510 may include an extremely large number of fabric devices, the query operations made by administration application 502 to obtain a list of the fabric devices may be made on a per host adapter port granularity or a set of host adapter ports
25 granularity so that the number of devices returned by the query may be a manageable number.

[0045] Turning now to Figure 6, a flowchart is provided illustrating a device discovery process according to one embodiment of the present invention. The discovery
30 process may be part of a host system reconfiguration boot process. Alternatively, if a

non-reconfiguration boot is performed of the host system in which device discovery is not performed, the discovery process may run when a previously created device node is accessed in the host system. This discovery process may also be performed if the host system's host adapter link is lost (e.g. cable pulled out).

5

[0046] During the discovery process a fabric driver is loaded as indicated at 602. Note that if the fabric driver is already loaded, then it may no be necessary to load the fabric driver again. The link state of each host adapter port may be checked as indicated at 604 and 606. If a port's link is down, the link may be designated as offline as indicated at 608, and the discovery process for that link may wait for a state change in the link to online, as indicated at 610. If the link is later restored, the discovery process may continue for that link.

[0047] If the link is up, the link type is determined, as indicated at 612. In one embodiment, a fabric login is attempted through the link. If the fabric login is successful, the link is determined to be a fabric link. If unsuccessful the link is determined to be a direct attach link. For a direct attach link, the one or more direct attach devices are discovered and brought online by creating operating system device nodes for each direct attach device. If the link is a fabric link, the link is designated as such, but no device discovery or onlining of devices is performed for the fabric link as part of this discovery process, as indicated at 616. The discovery process as illustrated in Figure 6 may be repeated any time a link goes out (e.g., cable pulled, power off, host reboot, etc.).

[0048] The device discovery process illustrated in Figure 6 provides for the discovery and onlining of direct attach devices since the discovery and onlining of such devices may be completed quickly due to the limited number of such devices. However, for a host system's fabric links, device discovery is not performed as part of the normal discovery process performed at reconfiguration boot up or the first time a node is attempted to be accessed after a link has been down and brought back up. Instead, fabric devices may be discovered using the on-demand node creation process described herein.

[0049] One embodiment of the on-demand node creation process is illustrated by the flowchart of Figure 3. A request may be received to discover fabric attached devices, as indicated at 310. In response to the request to discover fabric attached devices, a fabric
5 may be requested to identify fabric devices available to a host system, as indicated at 320. A list of identified fabric devices may be received from the fabric, as indicated at 330. Note that the term “list” simply refers to the information provided by the fabric driver and does not require any particular format.

10 [0050] This list may be provided to a user for selection of a subset of the fabric devices, as indicated at 340. Alternatively, selection of a subset of the fabric devices may be performed without user involvement, as indicated at 345. A request may be received to on-line the subset of identified fabric devices, as indicated at 350. In response to the request to on-line the subset of identified fabric devices, a node may be created for each
15 fabric device of the subset not already on-line, as indicated at 360. The node provides a mechanism for processes to communicate with the corresponding device from the host system. The on-line status on each fabric device of the subset may be stored, as indicated at 370.

20 [0051] Another embodiment of the on-demand node creation process is illustrated by the flowchart of Figure 7. An administration application may make a request for a list of fabric devices, as indicated at 702. This request may have been initiated by a system administrator using the administration application for the host system. In other embodiments, this process may be initiated automatically, for example in response to an
25 event (e.g. fibre channel protocol event) or a request from another application or process. A fabric driver may receive the request from the administration application as indicated at 704. The fabric driver may access a fabric name server to obtain the requested information and return the requested list of fabric devices to the administration application. Note that the term “list” simply refers to the information provided by the
30 fabric driver and does not require any particular format.

[0052] The administration application may receive the list and a subset of devices may be selected from the list, as indicated at 706. In one embodiment the list may be displayed through a graphical user interface to a system administrator. In other
 5 embodiments, a command line or textual user interface is used. The system administrator may select particular devices from the list to be onlined or offlined. The administration application may then call the fabric driver to online or offline devices as indicated by the selections made for the subset of devices from the list, as indicated at 708. The driver may then attempt to online the devices selected to be onlined (and may offline the devices
 10 selected to be offlined), as indicated at 710. Onlining a device may include the creation of an operating system device node for that device. The device node provides a mechanism for processes to communicate with the corresponding device from the host system.

[0053] A persistent repository may be updated (or created if not already existing) to indicate which devices are currently online, as indicated at 712. During operation, the persistent repository may be updated to reflect any changes in fabric devices. For example, if a device indicated by the persistent repository to be currently onlined is removed from the fabric the persistent repository may be updated accordingly as indicated
 20 at 714. Changes in the fabric may be detected by an event generated in the fabric and communicated to the administration application (or library) by the fabric driver.

[0054] Figure 8 illustrates a host system 800 coupled to a fabric 510 according to one embodiment of the present invention. The host system 800 includes a fabric driver 504
 25 for communicating with a fibre channel fabric. A library 503 may be provided as an interface between the administration application 502 and the fabric driver 504. The library may be made part of the operating system libraries and may be useable by other applications on the host system. For one embodiment of the present invention, fabric driver 504 may include various sub-modules. For example, in a fibre channel
 30 implementation, the fabric driver may include a fibre channel protocol driver module

1 (FCP) 802. The FCP module may be part of the operating system kernel and may
perform all protocol related operations required for fibre channel use on the host
operating system. For example, FCP module 802 may be a SCSI over fibre channel
encapsulation driver module for supporting SCSI over fibre channel. Driver 504 may
5 also include a transport layer 804. This module may perform all generic fabric
operations. In one embodiment transport layer 804 may include a fibre channel port (FP)
driver for each fibre channel port on the host system. Each FP driver may also be part of
the operating system kernel. The FP driver performs all generic fibre channel operations
such as topology discovery (e.g., loop, point-to-point, fabric, etc.), device discovery (on
10 various topologies), handling extended link services, handling link state changes, etc.
The fabric driver 504 may also include host adapter drivers 808 for each host
adapter/controller board on the host system. For example FCA drivers may be present for
fibre channel adapters having fibre channel ports on the host system.

15 **[0055]** Figure 9 illustrates a more detailed flowchart of the on-demand node creation
process for fibre channel host adapter ports of a host system. In one embodiment, the
host system may be executing a UNIX type operating system such as the Solaris™
operating system from Sun Microsystems. A Solaris operating system and fibre channel
fabric will be assumed by way of example in the following description.

20 **[0056]** A user (e.g. system administrator, other process, etc.) may launch or access the
administration application to obtain a list of devices available on a fabric port (e.g. FCA
port), as indicated at 902. For example, a system administrator may interface to the
administration through a graphical user interface (GUI), command line use interface, etc.
25 Through the user interface, the system administrator may query for devices available on a
single port or on a set of ports, as indicated at 904. On Solaris, for example, an FCA port
may have a file system representation of the form, e.g.,
/devices/pci@1f,0/pci@1/SUNW,qlc@4/fp@0,0:fc.

[0057] One or more library interfaces may be provided as an interface between the administration application and the fabric driver. This library interface(s) may be made part of the operating system's standard storage libraries. The administration application may then call one or more of the library interfaces to obtain the list of devices, as indicated at 906. These libraries may be shared between the administration application and other storage applications. The library calls into the fabric driver to obtain the list of devices, as indicated at 908.

[0058] In one embodiment, the library calls into an FCP module using the standard driver IOCTL interface (the call may actually be made into an FP module which merely passes on the IOCTL call to the FCP module) to obtain the list of devices. The library may then wait for the FCP module to return the list. The FCP module may then make a call into a port driver module(s) to obtain the list of fabric devices available through the fabric port(s) on which the request was made, as indicated at 910. For example, the FCP module may call an FP module(s) using the FC transport APIs between ULPs and the FC transport to obtain the list of fabric devices available through the FCA port(s) on which the request was made.

[0059] The fabric driver (e.g. port driver module(s) of the fabric driver) then obtains the device information from the fabric. In one embodiment, the FP module(s) may query a NameServer on the fabric (the Name Server may be maintained by FC switches) to obtain the list of FC ports and return this list to FCP, as indicated at 912.

[0060] In one embodiment, the fabric driver may check the list for devices following a particular protocol and return a list of those devices to the administration application library interface. For example, the FCP module may check for SCSI ports from the received list (e.g. disks and tapes) and return the list of SCSI capable FC ports to the library, as indicated at 914. In another embodiment, the list may include selective LUNs (logical units) behind an FC port. The library may then pass this list on to the

administration application, as indicated at 916. The administration application may then display the list through its user interface, as indicated at 918.

[0061] Turning now to Fig. 10, a flow chart is provided illustrating a method for
5 online selected fabric devices on-demand from a list of fabric devices, according to one
embodiment. In one embodiment, a list of fabric devices may have been obtained by the
method illustrated in Fig. 9. A user (e.g. system administrator, other process, etc.)
chooses a subset of devices from this list to online and passes this list of devices to the
administration application, as indicated at 932. The administration application may then
10 call the fabric driver to online the selected devices. For example, the administration
application may call an FCP module (via a library interface) to online the selected
devices, as indicated at 934. The FCP module, using the services of FP module(s),
attempts to online each of the selected devices in the list and reports each successful
online to the library, as indicated at 936. For every device successfully onboarded by the
15 FCP module, the administration application library interface updates a persistent
repository to reflect devices currently onboarded, as indicated at 938. Note also that
offlining a node through the administration application may cause an event to be
generated to trigger an update the persistent repository.

[0062] Turning now to Fig. 11, one embodiment is illustrated of a mechanism for
20 dynamically updating the persistent repository to reflect changes that occur in the fabric.
A device state change may occur in the fabric, as indicated at 950. For example, a fabric
disk drive online for the host system may be removed from the fabric. An event may be
generated to indicate this change. The FCP module may inform the library/application of
such changes that effect the device list, as indicated at 952. The administration
25 application library interface may update the contents of the persistent repository in
response to receiving notification of a change, as indicated at 954, so that the persistent
repository is dynamically updated to reflect such changes. The persistent repository may
be stored, for example, on persistent storage such as a disk drive or non-volatile memory.

30

1
2
3
4
5
6
7
8
9
10

[0063] The persistent repository may allow a host's fabric device configuration to persist across reboots and shutdowns, as illustrated in Fig. 12 for one embodiment. On a host reboot (970), a component of administration application (e.g. in the Solaris 're' scripts) reads the persistent repository to determine which devices were previously online, as indicated at 972. The administration application then calls the fabric driver (via the library) to online the fabric devices that were onlined prior to the reboot, as indicated at 974. The fabric driver attempts to online each device indicated by the persistent repository and reports successful onlines to the library, as indicated at 976. The library may then update the persistent repository to reflect the currently online devices, as indicated at 978.

11
12
13
14
15

[0064] An example of an entry in a persistent repository is:
/devices/pci@1f,4000/pci@2/SUNW,qlc@4/fp@0,0:fc::50020f23000000e6.
In this example, the entry identifies the /devices path to an onlined FC port. In other embodiments, additional information may be included. For example, the repository may identify onlined fabric devices on a per LUN basis.

16
17
18
19
20
21
22
23
24
25
26
27
28
29

[0065] As discussed above, fabric devices may be efficiently configured for access through a host adapter port on a host computer by using an on-demand node creation process. Also, a persistent repository may store an indication of fabric devices that are online through each host adapter port. The persistent repository may be used to maintain the node configurations for each host adapter port across reboots and shutdowns. Thus, fabric devices may initially be configured, for example, through an administration application which initiates node configuration requests for selected fabric devices. If the host system is rebooted, the previous fabric device node configurations may be restored by accessing the persistent repository to determine the previously onlined fabric devices and bringing those devices online again.

30
31

[0066] Other changes may also occur in the fabric, or in the host system's connectivity to the fabric, for which it may be desirable to alter the fabric device node

configuration for the host system. It may be desirable to dynamically reconfigure a host's fabric device node configuration in response to fabric state changes as those changes are detected. For example, a host adapter providing the interface between a host computer and the fabric may become disconnected from the fabric. The host adapter may lose its connection to the fabric because of, for example, an adapter failure, or cable disconnection, or driver detachment, etc. Later, the host adapter may be dynamically restored during operation of the host computer. A mechanism may be provided to reconfigure (e.g. perform node creation for) the fabric devices that were online through the host adapter prior to its restoration.

10

[0067] Another dynamic fabric state change that may occur is the addition of a new device to the fabric visible through a host adapter port of the host computer. A host computer may desire to be able to access all newly added fabric devices through a particular host adapter port. A dynamic node creation mechanism may be provided to dynamically configured newly connected fabric devices through a host adapter port.

15

[0068] Turning now to Figure 13, a diagram is provided illustrating components for providing dynamic node reconfiguration for a host system 1300 in response to fabric state change events. The host system 1300 may have one or more ports 1302 for coupling to a fabric. A fabric event agent 550 may be provided for detecting fabric state changes and responding accordingly. Fabric event agent 550 may be a software module on the host system that interfaces with fabric driver 504 and persistent repository 506. In one embodiment, fabric event agent 550 may be part of the operating system libraries 503 as discussed in connection with Figure 8.

20

[0069] Fabric event agent 550 may receive events from fabric driver 504 indicating a state change in the fabric connectivity. In one embodiment fabric event agent 550 registers with a system event framework of the host operating system to listen to events related to one or more fabric adapter ports of the host system.

25

30

[0070] One type of event monitored by fabric event agent 550 may be events indicating changes in the connectivity of a host adapter port to the fabric. For example, an event may be generated through or by fabric driver 504 if a host adapter port loses connection to a fabric. A host adapter port may lose connection to the fabric for a variety of reasons. One such reason may be a physical disconnection of the cable or interconnect to the host adapter. Another reason may be a failure in the host adapter. A host adapter may be an I/O system board, host bus adapter card, or other circuitry on or connected to the host system. If, for example, a host bus adapter card fails, the failure may be detected by the fabric driver and an event may be generated indicating that the host adapter is down. In some embodiments an event indicating that a host adapter port is no longer available may be generated when the driver for that host adapter (e.g. FCA driver 808, see Figure 8) detaches. A host adapter driver may detach upon detecting a host bus adapter card failure or may be manually detached (e.g. disabled). Thus, an event may be generated to indicate the loss of host adapter connectivity to the fabric for a variety of different reasons.

[0071] If fabric event agent 550 receives an event indicating a loss of host adapter connectivity for one or more host adapter ports, the fabric event agent 550 may take appropriate action. For example, those fabric devices which were onlined through the lost host adapter may be unconfigured. Fabric event agent 550 may request an unconfigure operation for the fabric devices that were configured or onlined through the lost host adapter port or ports. The fabric event agent 550 may access the persistent repository 506 to determine which fabric devices were onlined through the lost host adapter (or host adapter port). The unconfigure operation may disable the operating system nodes for the fabric devices that were configured through the lost host adapter. It may be desirable to disable these nodes so that other processes or application executing on the host system do not attempt to use fabric devices that are no longer available from the host system.

[0072] When a host adapter is restored, an event may be generated to indicate that the host adapters connectivity to the fabric is now available. In one embodiment, such an event may be generated upon host adapter driver attach. In response to such an event, fabric event agent 550 may access persistent repository 506 to determine fabric devices that were previously online for the now-restored host adapter. Fabric event agent 550 may request a configuration operation (node creation) for the fabric devices that were previously online for a particular host adapter or host adapter port. Thus, a dynamic node creation mechanism is provided to respond to the restoration of connectivity to the fabric through a host adapter. If a host adapter is replaced, repaired, cable reconnected, driver reattached, etc., the fabric device nodes that had previously existed in regard to that host adapter may be recreated. The reconfiguration operation to recreate a previous node configuration for a restored host adapter may involved verifying the fabric devices indicated by the persistent repository for the restored host adapter, and then performing node creation for the verified fabric devices.

[0073] Another type of fabric connectivity state change that may be detected by the fabric event agent 550 is an event indicating a newly connected fabric device visible through a particular host fabric adapter. A host system may have a policy of configuring any newly connected fabric device for a particular host fabric adapter. In some embodiments a host may have a policy of configuring any newly connected fabric device of a particular type, such as storage devices, visible through a particular host adapter. If fabric event agent 550 receives an event indicating a newly connected fabric device through a host adapter, the fabric event agent 550 may request a configuration operation for the newly connected fabric device. The configuration operation may be a node creation operation to create an operating system node to make the newly connected fabric device accessible for the host system through the host adapter. The persistent repository 506 may be updated to indicate that the newly connected fabric device is online through the particular host adapter.

[0074] For one embodiment of the present invention, fabric driver 504 may include

various sub-modules as illustrated in Fig. 13A. For example, in a fibre channel implementation, the fabric driver may include a fibre channel protocol driver module (FCP) 802. The FCP module may be part of the operating system kernel and may perform all protocol related operations required for fibre channel use on the host operating system. For example, FCP module 802 may be a SCSI over fibre channel encapsulation driver module for supporting SCSI over fibre channel. Driver 504 may also include a transport layer 804. This module may perform all generic fabric operations. In one embodiment transport layer 804 may include a fibre channel port (FP) driver for each fibre channel port on the host system. Each FP driver may also be part of the operating system kernel. The FP driver performs all generic fibre channel operations such as topology discovery (e.g., loop, point-to-point, fabric, etc.), device discovery (on various topologies), handling extended link services, handling link state changes, etc. The fabric driver 504 may also include host adapter drivers 808 for each host adapter/controller board on the host system. For example FCA drivers may be present for fibre channel adapters having fibre channel ports on the host system.

[0075] Figure 14 is a flow chart illustrating an event driven node configuration process in response to dynamic changes in fabric connectivity. A fabric driver on a host system (e.g. fabric driver 504) may detect fabric state changes, as indicated at 1402. As described above, fabric state changes may include changes in the connectivity of a host adapter or host adapter port to the fabric and/or changes in the availability of fabric device through a particular host adapter or host adapter port. The fabric driver may generate an event to indicate the fabric state change, as indicated at 1404. A fabric event agent, registered to receive such events, may receive the event and determine the event type as indicated at 1406. The event agent may request a configuration/unconfiguration operation in response to the event, as indicated at 1408. The configuration/unconfiguration operation may depend on the event type, and the persistent repository may be accessed to be updated to reflect fabric device node changes or to determine which fabric devices are to be configured/unconfigured.

30

[0076] In one embodiment the fabric event agent 550 may access a host adapter configuration file to determine the action to be taken in response to a fabric state change event. Figure 15 illustrates a system including a host adapter configuration file 552. The host adapter configuration file 552 may be stored in a manner accessible by the host system. In one embodiment, a host adapter configuration file 552 may be provided for each host adapter of the host system. In other embodiments, a single host adapter configuration file may provide information for each host adapter of the host system. The host adapter configuration file 552 may indicate the action to be taken when a fabric state change event is detected for a particular host adapter.

[0077] In one embodiment, host adapter configuration file 552 comprises one or more user-defined attributes for a host adapter. The attribute may indicate whether or not fabric devices on that particular fabric host adapter should be configured or unconfigured upon detection of a particular fabric state changed event. For example, if an event is detected indicating that a particular fabric host adapter has been disconnected from the fabric, an attribute in the host adapter configuration file 552 for that host adapter may indicate that the fabric devices configured with nodes for that particular host adapter should be unconfigured (e.g. off-lined). The host adapter configuration file 552 may also include an attribute indicating the action to be taken when a particular host adapter's connectivity is restored. For example, an attribute in the host adapter configuration file may indicate either to configure or not to configure the fabric devices on a particular fabric host adapter upon an event indicating reattachment of that host adapter. The configuration file 552 may also include attributes indicating whether or not newly connected fabric should be dynamically configured for a particular host adapter. In other embodiments, an attribute may be included in the configuration file 552 to indicate whether or not fabric devices should be unconfigured upon detection of an event indicating that the device is no longer available on the fabric. In some embodiments, attributes may indicate whether or not a particular type (e.g. storage device or hard drive) of fabric device should be configured/unconfigured upon detection of an event indicating that the particular type of device has been added/removed to/from the fabric visible

through a particular host adapter or host adapter port. In one embodiment, one attribute may indicate whether or not to perform the configure/unconfigure operation and another attribute may indicate the device type.

- 5 **[0078]** An example host adapter configuration file format is as follows:

Host Adapter Port	Configure on Attach	Unconfigure on Detach	Configure Newly Connected Device
/pci@4,2000/pci@4/SUNW,qlc@4/fp@0,0	Yes	Yes	Yes
/pci@4,2000/pci@4/SUNW,qlc@5/fp@0,0	Yes	Yes	No
/pci@4,4000/pci@4/SUNW,qlc@4/fp@0,0	Yes	Yes	Yes
/pci@4,4000/pci@4/SUNW,qlc@5/fp@0,0	Yes	Yes	No

- 10 **[0079]** The Host Adapter Port column may identify a particular host adapter or host adapter port. The Configure on Attach column indicates whether or not fabric devices on a particular host adapter (port) should be configured when an event indicating fabric connectivity for that host adapter (port) is detected. The Unconfigure on Detach field indicates whether the devices configured on that fabric host adapter (port) should be unconfigured or not when a fabric host adapter (port) is detached. The Configure Newly Connected Device field indicates if a newly connected fabric device is to be configured or
- 15 not on a particular host adapter (port). Fewer or more fields may be included in the host adapter configuration file, and the other fields may indicate other actions to be taken in response to dynamic fabric state change events. The fields or attributes in the host adapter configuration file may be set by a user, such as a system administrator. Thus allowing the user to specify the action to be taken in regard to certain fabric connectivity
- 20 state changes.

- 25 **[0080]** Turning now to Figure 16, a flowchart is provided illustrating an event driven configuration process for an event in which a host adapter loses connectivity to the fabric. When a host adapter loses fabric connectivity, a corresponding event may be generated, as indicated at 1602. In response to this event, the configuration file for the host adapter

may be read, as indicated at 1604, to determine the action to be taken. The host adapter configuration file may indicate whether or not fabric devices online through the disconnected host adapter should be unconfigured or offlined. If the host adapter configuration file indicates that the devices are to be unconfigured, an unconfiguration operation may be requested. The persistent repository may be accessed to determine which fabric devices are online for the disconnected host adapter or host adapter port, as indicated at 1606. An unconfiguration operation may then be requested for each of the fabric devices online for the disconnected host adapter, as indicated at 1608. The unconfiguration operation may involve offlining the fabric devices for the disconnected host adapter. Offlining the devices may include disabling or deleting the operating system nodes for those fabric devices. In one embodiment receiving the event (1602), reading the host adapter configuration file (1604), accessing the persistent repository (1606), and requesting the unconfiguration operation (1608) may be performed by a fabric event agent. The fabric event agent may be part of an operating system library.

15

[0081] Turning now to Figure 17 a flowchart is provided illustrating an event driven configuration process for an event in which a host adapter acquires or regains connectivity to the fabric. When a host adapter acquires fabric connectivity, a corresponding event is generated as indicated at 1702. In response to this event the configuration file for the host adapter may be read, as indicated at 1704, to determine the action to be taken. The host adapter configuration file may indicate whether or not fabric devices previously online through the reconnected host adapter should be configured (e.g. online). If the host adapter configuration file indicates that the devices are to be configured, a configuration operation may be requested. The persistent repository may be accessed to determine which fabric devices were previously online for the connected host adapter, as indicated at 1706. A configuration operation may then be requested for each of the fabric devices previously online for the disconnected host adapter, as indicated at 1708. In one embodiment, a verification operation may first be performed for each previously online device to verify that the device is still available on the fabric. To verify device availability, a fabric event agent may request a verification operation from a

30

fabric driver. The fabric driver may access the fabric name server to obtain the fabric device list to verify that the device is still available. The configuration operation may involve onlining the fabric devices for the connected host adapter. Onlining the devices may include creating operating system nodes for those fabric devices. In one embodiment receiving the event (1702), reading the host adapter configuration file (1704), accessing the persistent repository (1706), and requesting the unconfiguration operation (1708) may be performed by a fabric event agent. The fabric event agent may be part of an operating system library.

10 **[0082]** If the no fabric devices were previously online for the connected host adapter port, or if the persistent repository contains no entries for the connected host adapter port, an initial configuration process may be initiated. The initial configuration process may be an on-demand node configuration process such as described in conjunction with Figs. 7-10 above. For example, a list of fabric devices for the connected host adapter port may be
15 passed to an administration application and a user may be prompted to select a subset of devices to be onlined.

[0083] Figure 18 illustrates a flowchart for an event driven configuration in regard to a fabric state change of the connection of a new fabric device or restoration of a
20 previously connected fabric device. In some embodiments, a host system may have a policy of onlining all newly connected fabric devices for one or more host adapter ports. In other embodiments such a policy may be in regard to only certain types of newly connected fabric devices, such as storage devices. The addition of a newly connected or restored fabric device may be detected by a fabric driver for one or more host adapter
25 ports on a host system. The fabric driver may generate an event indicating the addition of a device to the fabric, as indicated at 1802. In response to the event, the persistent repository may be accessed to determine if an entry already exists in the persistent repository for the added fabric device, as indicated at 1804. An entry may already exist, for example, if a previously available fabric device had been removed from the fabric and
30 then restored. If an entry already exists, indicating that the added device should be

configured, then a configuration operation may be requested, as indicated at 1806 and 1814.

[0084] If an entry does not exist in the persistent repository for the added fabric device, the host adapter configuration file may be accessed to determine the action to be taken, as indicated at 1806 and 1808. The host adapter configuration file may indicate whether or not the newly connected fabric device should be online for the host adapter port corresponding to the event. If indicated by the configuration file, a configuration operation may be requested to online the newly connected fabric device, as indicated at 1810. The persistent repository may be updated to indicate that the newly connected fabric device is online for the host adapter port, as indicated at 1812. In some embodiments, receiving the event (1802), accessing the persistent repository (1804) accessing the host adapter configuration file (1808), requesting the configuration operation (1810, 1814), and/or updating the persistent repository (1812), may be performed by a fabric event agent. In some embodiments the configuration operation includes a node creation process as described above. The newly connected fabric device may be verified before performing node creation. The fabric device may be verified by requesting a list of fabric devices from the fabric name server as described above.

[0085] A similar process to that described for Figure 18 may be performed for events indicating the disconnection of a fabric device. Other fabric state change events may be handled as well.

[0086] In one embodiment the fabric state change event may be delivered by the fabric driver through an event abstract data structure, such as:

```
event id
event type: attach/detach/fabricchange
event change type: insert/remove/null
event port: the host adapter port which has the change occur, or global
```

[0087] The event id provides a unique identification for each event. The event type indicates the type of event delivered, e.g. the type of event detected by the fabric driver. For example, an event type of “attach” may indicate that a host port adapter driver or instance of a driver has just been dynamically attached for a particular host adapter port, indicating that the host adapter port is now connected to the fabric. Similarly an event type of “detach” may indicate that a host port adapter driver or instance of a driver is no longer attached, indicating that host adapter port is no longer connected to the fabric. The event type “fabric change” may indicate that the fabric driver has received a configuration change notice from the fabric. The event data structure may include an event change type field to indicate the type of fabric change. The event change type field of the event data structure may be valid for only the fabric change type of event. The event change type field may indicate a type of “added” when a fabric device has been added or “removed” when a fabric device has been removed. In one embodiment, the event change type field may indicate a null value. If the value is null, the fabric event agent may determine the change type by requesting a current fabric device list (e.g. from the fabric driver/ fabric name server). The fabric event agent may compare the current fabric device list to a previous device list (or to the persistent repository) to determine if a fabric device has been added or removed from the fabric.

[0088] The event port field of the event data structure indicates for which fabric host adapter port the state change has occurred. In one embodiment an event may be a global event, in which case the fabric event agent may perform the dynamic reconfiguration process described herein for each host adapter port.

[0089] Various modifications and changes may be made as would be obvious to a person skilled in the art having the benefit of this disclosure. Note also that the flow charts described herein do not necessary require a temporal order. It is intended that the following claims be interpreted to embrace all such modifications and changes and, accordingly, the specifications and drawings are to be regarded in an illustrative rather than a restrictive sense.

[0090] Various embodiments may further include receiving, sending or storing instructions and/or data implemented in accordance with the foregoing description upon a computer readable medium. Generally speaking, a computer readable medium may include storage media or memory media such as magnetic or optical media, e.g., disk or CD-ROM, volatile or non-volatile media such as RAM (e.g. SDRAM, DDR SDRAM, RDRAM, SRAM, etc.), ROM, etc. as well as transmission media or signals such as electrical, electromagnetic, or digital signals, conveyed via a communication medium such as network and/or a wireless link.

10

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217